

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2003-114817

(43)Date of publication of application : 18.04.2003

(51)Int.Cl.

G06F 12/00  
G11B 7/007  
G11B 20/12  
H04N 5/85

(21)Application number : 2001-342145

(71)Applicant : TOSHIBA CORP

(22)Date of filing : 07.11.2001

(72)Inventor : ANDO HIDEO  
TAKAHASHI HIDEKI  
MIMURA HIDENORI

(30)Priority

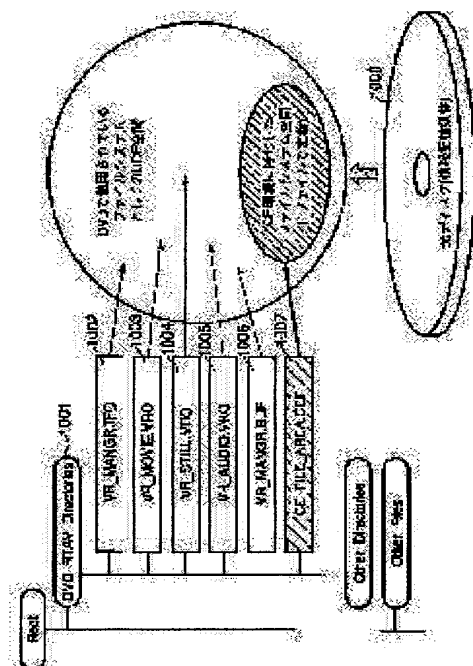
Priority number : 2001236731 Priority date : 03.08.2001 Priority country : JP

## (54) INFORMATION STORAGE MEDIUM AND INFORMATION RECORDING METHOD

## (57)Abstract:

PROBLEM TO BE SOLVED: To provide an information storage medium which can make use of individual advantages of a plurality of file systems.

SOLUTION: An information storage medium (1) has a logical space managed by a predetermined file system, wherein the logical space is managed by either one of two different types of file systems and a file (1007) to designate a space area of the other file system is provided in the logical space.



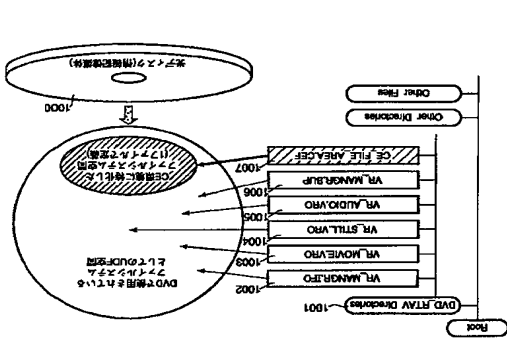
(19)日本特許庁(JP) (12) 公開特許公報 (A) (11)特許出願公開番号  
特開2003-114817  
(P2003-114817A)  
(43)公開日 平成15年4月18日(2003.4.18)

(51)IntCl. <sup>7</sup>		識別記号		FI	
G 0 6 F	12/00	5 2 0		G 0 6 F	12/00
G 1 1 B	7/007	20/12		G 1 1 B	7/007
H 0 4 N	5/85			H 0 4 N	5/85
審査請求 有		請求項の数 2 OL (全 23 頁)		特許請求の範囲	

(21)出願番号	特願2001-342145(P2001-342145)	(71)出願人	00003078 株式会社東芝
(22)出願日	平成13年11月7日(2001.11.7)	(72)発明者	安東 秀夫 神奈川県川崎市幸区御門70番地 株式会社 東芝御町事業所内
(31)優先権主張番号	特願2001-236731(P2001-236731)	(72)発明者	高橋 秀樹 神奈川県川崎市幸区御門70番地 株式会社 東芝御町事業所内
(32)優先日	平成13年8月3日(2001.8.3)	(74)代理人	100058479 弁理士 鈴江 武彦 (外6名)
(33)優先権主張国	日本(JP)		

(54)【発明の名称】 情報記憶媒体及び情報記録方法

(57)【要約】  
【課題】複数のファイルシステムの夫々の長所を生かすことが可能な情報記憶媒体を提供すること。  
【解決手段】所定のファイルシステムで管理される論理空間を有する情報記憶媒体(1)であって、異なる2種類のファイルシステムのどちらから一方のファイルシステムで前記論理空間が管理され、この論理空間上に他方のファイルシステムの空間領域を指定するためのファイル(10007)を有する。



(2) 特開2003-114817  
格により定義されているファイルだけでなく、ワープロや表計算ソフトのファイルなどを一つのディレクトリで統一的に扱う環境を提供することが望まれている。AV専用のファイルシステムではPCファイルの取り扱いが煩雑になる。  
【0008】この発明の目的は、上記問題を解決するためになされたものであり、複数のファイルシステムの夫々の長所を生かすことが可能な情報記憶媒体及び情報記録方法を提供することにある。  
【0009】つまり、同一の情報記憶媒体上でCE市場に特化した簡単な構造を持つファイルシステムの採用が可能とし、同時に同じ情報記憶媒体上で既存のDVDアプリケーションファイル及びPCファイルの取り扱いが可能とする。  
【0010】  
【課題を解決するための手段】上記課題を解決し目的を達成するために、この発明の情報記憶媒体及び情報記録方法は、以下のよう構成されている。  
【0011】(1)この発明の情報記憶媒体は、所定のファイルシステムで管理される論理空間を有し、異なる2種類のファイルシステムのどちらから一方のファイルシステムで前記論理空間が管理され、この論理空間上に他方のファイルシステムの空間領域を指定するためのファイル(10012) (2)この発明の情報記録方法は、所定のファイルシステムで管理される論理空間を有する情報記憶媒体に対して、異なる2種類のファイルシステムのどちらから一方のファイルシステムで前記論理空間を管理し、この論理空間上に他方のファイルシステムの空間領域を指定するためのファイル(10013) (3)AV特化のファイルアプリケーションに関して強い制限を設けることでCEの負荷を軽減する。  
【0014】(1)UDFとCE市場に特化した独自のファイルシステムと異なる2種類のファイルシステムを同一情報記憶媒体上に共存記録させる。  
【0015】(2)異なる2種類のファイルシステムのどちらから一方のファイルシステムで情報記憶媒体上の論理空間を管理させ、その論理空間上に他方のファイルシステム空間領域を指定するためのファイル(10016) (3)AV特化のファイルアプリケーションに関して強い制限を設けることでCEの負荷を軽減する。  
【0017】(4)UDFのファイルシステムドライバを持たなくてもUDF上のファイル操作が可能な仕組みを得る。  
【0018】以下、この発明の実施の形態について図面を参照して説明する。  
【0019】図1は、CE環境に特化したファイルシステムとDVDで使用されているUDFとの間の共存方法

を示している。光ディスク（情報記録媒体）1000上のDVD用のファイルはUDFに割り当てられたディレクトリに格納される。ここでは記録系アプリケーションファイルが格納するために定義されたDVD\_RTA V Directories1001に複数のファイルが置かれていた状態を示している。

【0020】この実施例では従来のV i d e o 記録用ファイル群（VR\_MANGR、IFO 1002、VR\_MOVIE、VR\_O 1003、VR\_STILL、VR\_O1004、VR\_AUDIO、VR\_O 1005、VR\_MANGR、BU P1006）に加え新規に定義されるファイルC E \_F I L E \_A R E A、C E F 1007が置かれている。従来のV i d e o 記録用ファイル群はUDFで直接管理される。それに対して、C E \_F I L E \_A R E A、C E F はファイルの占有するエリアはUDF上に定義されるもののUDFとは別の領域に特化したファイルシステム空間を内部に持っていることを特徴とする。このC E \_F I L E \_A R E A、C E F はファイルではあるが1つのパーティションのような機能を持つ。すなわち、このファイルはコンテンツサイズに応じ変わるのではなく、まずは領域として確保されるだけである。この領域確保の時にUDF上にファイル登録される必要があり、いったんUDF上にファイル登録された後は、このファイル内の情報を書き換えてもファイルサイズ及び占有しているアドレス位置は変化する。すなわち、UDF上は不変の領域となる。もし、ファイルサイズを変更する場合には以降に示すように特別のプロセスを経ることが必要となる。

【0021】図2は、図1で示した方法において、UDF論理アドレス空間上で各種ファイルの配置設定方法を比較したものである。まず、PCファイルとDVDアプリケーションの管理ファイルについて、ファイルアロケーション上はセクタ単位の割り当てとする。この例のセクタサイズは2KBである。次にDVDオブジェクトファイル（画像用ファイル）は不連続部分を含有する連続領域であるCDA（Contiguous Data Area）の最小サイズが2MBであり、CDA中に欠陥セクタ等が含まれる場合には連続性がとぎれてしまう。このため、再生時のバッファアランダーランを防止するためにCDAサイズを調整してしまふ場所は、欠陥セクタ、ゾーンバウンダリ、他のファイルが入る、場合がある。CDAサイズを計算するためにはこれらの不連続点の数（セクタ数）と分布を考慮することが必要である。記録時に新たに発見された欠陥が無い場合には、記録位置に関する情報をホストにあらためて入力できることでCDAサイズを計算することが可能である。記録時に新たに発見された欠陥をスキップする場合には発見された時点で必要十分なCDAサイズを求めるためには再計算が必要となる。このようにリアルタイム処理の方法についてはセットの作り方

E+L)、AD (L、F)、AD (L、F+2L)) CDAサイズとExtentサイズは等しく、全てのCDAサイズが一定（4MB以上）。CE環境に特化したファイルシステム空間内での疑似ファイル（AV\_FILE\_01、MPG/AV\_FILE\_02、MPG）サイズは、CDAサイズの整数倍とし、ディスク上の配置場所はCDAの配置場所と一致させる。AV疑似ファイルは固定長CDA単位での追加/解放を行う。

【0026】図5は、図1に示した方法での各種ファイルと疑似ファイルの表示解説図である。ここに示したようにC E \_F I L E \_A R E A、C E F については、内部はファイル構成をとっており、PC上でこのファイル内のCEファイル群を見るためには、特別のファイルV i e w e r を介する必要がある。また、このようにC E \_F I L E \_A R E A、C E F 内はファイル群で構成されているので、必要があれば、各ファイルをUDFから直接見える領域に取り出すことが可能である。一度UDFから直接見える場所に取り出されたファイルなら通常のPC上のファイル操作コピー、移動操作することが可能となる。また、逆にC E \_F I L E \_A R E A、C E F に戻すことも可能である。基本的に、シームレス再生を行うためには、C E \_F I L E \_A R E A、C E F が必要であるが、ディレクトリ上のアロケーションに工夫に取り込む時点で、特別なアロケーションツール介を行うことになる。

【0027】ここで、CE環境に特化したファイルシステム及びUDFの両者のメリットの比較、及び同一記憶媒体上に両者を共存させる意義についてまとめる。

【0028】CE環境に特化したファイルシステムを採用するメリットとしては以下のようなのが挙げられる。  
【0029】（1）使用目的をAV情報記録に特化させ、最適で簡素なファイルシステムにして、制御ソフトウェアのコンパクト化を図ることができる。このファイルシステム内でのPCファイルの混在記録を禁止し、制御ソフトウェアを簡素化する。

【0030】（2）CE環境に特化し、コンパクトなファイルシステムコンポーネントを構築し、上位レイヤーでの総合活用を可能にする。上位レイヤーでのPCファイルとの共存が可能とする。

【0031】ファイルシステムとしてUDFを採用するメリットとしては以下のようなのが挙げられる。

【0032】（1）現行DVD規格に基づいて作成するAV情報の記録が可能となる。

【0033】（2）PCファイルの混在記録が可能となる。

【0034】（3）既に製品化されたDVD関連製品のファイルシステムに関する制御部の有効活用が可能となる。

【0035】同一記憶媒体上に両者を共存させることにより、両者のメリットを引き出すことができる。

【0036】C E \_F I L E \_A R E A、C E F をアロケートするときの制限として次のようなものが考えられる。

【0037】（1）再配置を禁止する  
例えば、ファイルシステムレベルでデフラグなどを行うと配置がずれてしまうので、ファイルタイプ249あるいはDVD専用に新規番号の割り当て、Non-Relocatableに設定しておくなど、UDF上の特別な管理が必要である。UDFに対して再配置が禁止されれば、割り当てられた領域が変化しないのでLSN (Logical Address Number) などファイルとして割り当てられている領域情報をファイル内に持つて、その中でだけ処理することが可能となる。

【0038】（2）ファイルのスタートアドレスを固定にする

ファイルのスタートアドレスを固定にすることができればUDFを解釈することなく、CEは記録再生処理を行うことができる。

【0039】（3）C E \_F I L E \_A R E A、C E F の構成を示すファイルを別定義して配置  
全E x t e n t の位置情報を1ファイルに集めて置く。

【0040】図6は、C E \_F I L E \_A R E A、C E F のファイルサイズとの拡大（縮小）例である。ファイルは、1つ以上の領域によって構成されており、それぞれは、固定長1の倍数を単位としてサイズが可変になっている。

【0041】図7は、UDFによる従来のV o l u m e & F i l e S t r u c t u r e（ボリューム&ファイルストラクチャー）の読み出し手順である。これに対し、図8で示したようにC E \_F I L E \_A R E A、C E F を適用に設定することでUDFの解釈を大幅に減らし、V o l u m e & F i l e S t r u c t u r e を読み出すことが可能となる。ここでは、固定LSNに配置したT r i g g e r（トリガー）ファイル定義すること、このT r i g g e r ファイルをサーチしてUDFの解釈を不要にした例を示している。T r i g g e r ファイルとは、図10に示したようにC E \_F I L E \_A R E A、C E F の位置情報を示したものである。このT r i g g e r ファイルを定義することにより装置はUDFを解釈しなくてもファイルの位置が特定できる。T r i g g e r ファイルの解釈は必要であるが、UDFを解釈することに比べて遙かに容易である。

【0042】図11～図13にT r i g g e r ファイルの活用例を示した。例えばT r i g g e r ファイルであるか否かを確認するために定義している識別子256バイトがあれば識別誤りが発生する確率が十分小さい。このバイト数は必要に応じて小さくしてもよい。T r i g g e r ファイルはあらかじめ決めた固定LSNに配置されておりそこからMセクタ分をT r i g g e r ファイル

領域と決めておき、Triggerファイルの内容を複数回、異なるECCブロックに配置しておけば何らかの障害が発生してファイルの先頭領域が読み出せなくなっても情報取り出しが可能である。

【0043】図11に示すように、Triggerファイルの位置が確認される。まず、固定LSNからはじめのセクタが読み込まれる(ST11)、識別子(RBP0～255)が調べられる(ST12)、Triggerファイルであることが確認されると(ST13)、分割領域数(RBP256～257)が確認され(ST14)、分割領域の位置が確認される(ST15)。必要に応じてセクタ目以降が読み込まれる(ST16)。

【0044】続いて、図12を参照して、はじめのセクタが読み出せない場合には、次のECC(Error Correction Code)ブロックの先頭セクタが読み出される(ST21)。規定のアドレスを超える場合には(ST22、YES)、エラーとなる(ST23)。規定のアドレス以下であれば(ST22、NO)、識別子(RBP0～255)が調べられる(ST24)、Triggerファイルであることが確認されると(ST25、YES)、分割領域数(RBP256～257)が確認され(ST26)、ファイルの位置が確認される(ST27)。必要に応じてセクタ目以降が読み込まれる(ST28)。

【0045】続いて、図13を参照して、領域の追加削除を行う場合の処理について説明する。追加削除可能な単位に注意してUDF上で領域追加削除が処理され(ST31)、これにあわせてTriggerファイルが変更される(ST32)。

【0046】図39では、CE\_FILE\_AREA、C\_EFの割り当てとWRITE/READ処理方法について示している。CE\_FILE\_AREA、CEFははじめにUDFでLBN(Logical Block Number)内に割り当てられる。割り当てられた後、ファイルの位置情報をTriggerファイルに記録する。この例ではTriggerファイルもUDFに配置されたファイルであり、Triggerファイルの置かれていた位置はあらかじめ決められた固定LSNから始まるあらかじめ決められたMセクタ分である。このようにいったんTriggerファイルが生成されれば、CE装置ではUDFを解除するのではなくTriggerファイルからCE\_FILE\_AREA、CEFを記録して(ST91)、アプリケーション内で使われるアドレス(ADAP: Address in Application)を決定し(ST92)、このADAPをLSNに変換し(ST93)、変換されたLSNによりWRITE/READが可能となる(ST94)。

【0047】自由度を大きくするためにTriggerファイルは実体のファイルであるCE\_FILE\_AR

EA、CEFから分離した。しかし、CE\_FILE\_AREA、CEF内(例えば先頭)にTriggerファイルに記録される内容があってもよい。その場合CE\_FILE\_AREA、CEFの先頭LSNが固定であればCE装置にとって有利である。

【0048】上記の仕組みをシステムとして構成した例が図14である。ホストではCE\_FILE\_AREA、CEFを扱うためのLocal FSドライバを特許ライブラリコマンド制御部を介してドライバをコントロール可能である。例えば、はじめにディスクの全領域をCE\_FILE\_AREA、CEFとしてしまえばそれ以降全くUDF管理を行うことなく簡易なシステム制御が可能となる。

【0049】また、CE\_FILE\_AREA、CEFをDVD、RTAV Directories1001全体に予めアサインすることにより、UDFの変更を必要にすることもできる。

【0050】以下、図17～図19に示すフローチャートを参照して、CE\_FILE\_AREA、CEF空間に対する記録、CE\_FILE\_AREA、CEF空間の追加、CE\_FILE\_AREA、CEF空間の縮小などについて説明する。

【0051】最初に、図17に示すフローチャートを参照して、CE\_FILE\_AREA、CEF空間への記録処理についてまとめ、まず、Triggerファイルの参照の上、未記録CDAが探され(ST41)、未記録CDAに対してスタポートポイントがセットされ(ST42)、記録が開始される(ST43)。未記録CDAの残容量が所定容量を下回ると(ST44、NO)、追加処理(Triggerファイル再設定)が実行される(ST46)。記録が継続されている間(ST45、YES)、ST44～ST46の処理が繰り返される。

【0052】続いて、図18に示すフローチャートを参照して、未記録CDAの追加処理をまとめ、まず、アプリケーションエンジンからUDFドライバに対して追加が指示される(ST51)。これに伴いファイルの実体が拡張される(ST52)。つまり、Triggerファイルが変更される。

【0053】続いて、図19に示すフローチャートを参照して、CE\_FILE\_AREA、CEFのサイズ縮小処理について説明する。CE\_FILE\_AREA、CEFのサイズの縮小は、例えば、UDF上のデータが満杯になったときに実行される。まず、アプリケーションエンジンからUDFドライバに縮小が指示される(ST61)。これに伴いCE\_FILE\_AREA、CEFが解放され空きが調査され(ST62)、ファイルの実体が縮小される(ST63)。つまり、Triggerファイルが変更される。

【0054】図15には異なる本発明の他の実施例を説明する。図1ではUDF空間をベースとし、その

中にファイル定義によりC区領域に特化したファイルシステム空間を構築した。それに対して図15では光ディスク(情報記憶媒体)上の全論理空間を“CE領域に特化したファイルシステム空間”に割り当て、CE領域に特化したファイルシステム空間をベースにする。そして“CE領域に特化したファイルシステム空間”内に例えば“UDF\_FILE\_AREA、UDF”と言う名のファイルを定義して、そのファイルの配置場所としてDVDで使用されているファイルシステムとしての“UDF空間”を割り当てる。

【0055】図15に示した方法での各ファイルの記録方法を図16に示した。図15の方法を採用した場合に図16に示すように、光ディスク(情報記憶媒体)上の全論理アドレス空間上に固定サイズ(例えば4MB以上の)CDAで等分割し、記録ファイルの配置場所をそのCDA位置を基準として割り当てる所に特徴がある。すなわちUDF空間を割り当てるためのファイルである“UDF\_FILE\_AREA、UDF 2006”の位置とサイズはこの固定サイズで有る1個のCDAまたは複数のCDAを割り付ける。

【0056】図16に示した実施例ではCDA#24～CDA#26の場所を割り当てている。すなわちこのCDA#24～CDA#26の範囲内がUDF空間として確立される。CDA#24～CDA#26の範囲内を更に2kB単位で分割してUDF空間内のアドレスである相対的なLBN(Logical Block Number)を割り当てる。DVDのObject File(例えば“VR\_MOVIE.VRO”ファイル)はこのUDF空間内には配置可能となる。図2に示したようにDVD Objectfileに対するCDAは2MB以上の可変(但し2kB単位での変更)サイズで割り当てられ、このファイルのFile Entry内のAllocation Descriptorで示されるExtentは上記の相対的なLBN(Logical Block Number)を用いて記述される。

【0057】CE領域に特化したファイルシステム空間上での一般ファイル位置も固定サイズのCDAを基準として設定される。図15における“AV\_FILE\_01.MPG 2004”と“AV\_FILE\_02.MPG 2005”がAVファイル(映像または音声情報)に関係したObject File)を意味し、“TOC\_FILE\_1”と“AV\_MANG.MNG 2002”がそのObject Filesに関する管理情報が記録された管理ファイルになっている。Object Fileで有る“AV\_FILE\_01.MPG 2004”及び“AV\_FILE\_02.MPG 2005”に対する属性情報が記録され、全体のTOC(Table of Contents)で有ったファイルが“AV\_MANG.MNG 2002”で有ったファイルの再生手順を示す)に関する情報が“TOC\_FILE、IFO 2001”内に記録されている。

【0058】ここで、上記したこの発明の構成及び作用

効果についてまとめ、

【0059】(1) この発明の情報記憶媒体は、所定のファイルシステムで管理される論理空間を有し、異なる種類のファイルシステムとのどちらか一方のファイルシステムで前記論理空間が管理され、この論理空間上に他のファイルシステムの空間領域を指定するためのファイルを有する。

【0060】(2) この発明の情報記録方法は、所定のファイルシステムで管理される論理空間を有する情報記憶媒体に対して、異なる2種類のファイルシステムのどちらか一方のファイルシステムで前記論理空間を管理し、この論理空間上に他方のファイルシステムの空間領域を指定するためのファイルを記録する。

【0061】上記構成により、下記の作用効果が得られる。

【0062】(1) 追記(境界位置の移動含む)処理事前に2種類のファイルシステム領域を設定するがファイルの追加を繰り返すと一方の領域が狭くなり、2種類のファイルシステム領域間の境界位置を移動させる必要が生ずる。本発明では情報記憶媒体上の論理空間を管理するファイルシステム側で他方のファイルシステム空間領域を指定するためのファイルサイズを変更するだけで容易に境界位置の移動が可能となる。つまり、追記(境界位置の移動含む)処理の容易性が高い。

【0063】(2) 既存DVDアプリケーション規格上のAVファイルとCE市場に特化した独自のファイルシステム上に構築されるアプリケーションファイル間の録再生

2種類のファイルシステム領域が完全に分離され、基本的に一方のファイルシステム上の記録、再生処理が他方のファイルシステムに影響を及ぼす事が内での両者間の記録再生の容易性が高い。

【0064】(3) PCアプリケーションファイル既存のDVD規格に準拠した情報記憶媒体(光ディスク)で実装されているようにUDFで定義された論理空間上にPCアプリケーションファイルを記録することは容易となる。

【0065】(4) 既存のライティングソフト既存のDVDプレーヤーやDVDレコーダーはUDF上で処理できるようにシステムアップされているので、UDF領域に属して既存のDVDライティングソフトで記録可能となる。

【0066】(5) 既存のファイルシステムドライバー既存のDVDプレーヤーやDVDレコーダーはUDF上で処理できるようにシステムアップされているので、UDF領域に属して既存の既存のファイルシステムドライバーをさせる。

【0067】(6) プロケーションロジック本発明では情報記憶媒体上の論理空間を管理するファイルシステム側で“ファイル位置指定”の形で他方のフ

イルシステム空間領域を指定するためアロケーション  
ジックが単純化される。

【0068】(7) ファイルシステムードライバの簡  
素化  
ファイルシステムードライバの簡素化を図ることがで  
きる。

【0069】(8) テスト  
現存のDVDプレーヤーやDVDレコーダーはUDF上  
で処理できるようにシステムアップされているので、U  
DF空間上でのテストソフトとして既存のソフトが使用  
できるのでUDF空間上でのテストが非常に容易とな  
る。

【0070】(9) 部分消去とFS管理  
アプリケーション上の部分消去とFS管理への連携容易  
性が確保されている。

【0071】次に、図20～図29を用いてファイルシ  
ステムの方式としてのUDFについて説明する。

【0072】

<<UDFの概要説明 (UDFとは何か) >>>  
<<UDFとは何か>>>UDFとは Universal Disk Fo  
rmat の略で、主にディスク状態情報記憶媒体における  
“ファイル管理方法に関する規約”を示す。CD-R  
M、CD-R、CD-RW、DVD-Video、DVD-R  
OM、DVD-R、DVD-RAMは“ISO966  
0”で規格化されたUDFフォーマットを採用してい  
る。

【0073】ファイル管理方法としては基本的にルー  
トディレクトリ (Root Directory) を割り持ち、ツリ  
ー状にファイル管理する階層ファイルシステムを前提  
としている。

【0074】ここでは主にDVD-RAM規格 (File  
System Specifications) に準拠したUDFフォーマッ  
トについての説明を行うが、この説明内容の多くの部分  
はDVD-ROM規格内容とも一致している。

【0075】<<UDFの概要>>

<情報記憶媒体へのファイル情報記録内容>情報記憶媒  
体に情報を記録する場合、情報のまとまりを“ファイ  
ル” (File Data) と呼び、ファイルデータ単位で  
記録を行う。他のファイルデータと識別するためファイ  
ルデータ毎に独自のファイル名が付加されている。共通  
な情報内容を持つ複数ファイルデータ毎にグループ化す  
るとファイル管理とファイル検索が容易になる。この複  
数ファイルデータ毎のグループを“ディレクトリ” (Di  
rectory) または“フォルダー” (Folder) と呼  
ぶ。各ディレクトリ (フォルダー) 毎に独自のディレ  
クトリ名 (フォルダー名) が付加される。更にその複数の  
ディレクトリ (フォルダー) を集めて、その上の階層の  
グループとして上位のディレクトリ (上位フォルダー)  
でまとめる事が出来る。ここではファイルデータとディ  
レクトリ (フォルダー) を総称してファイル (File)

Descriptor 文で並記して有る。

【0093】この Sub Directory 402 の File Identifier Descriptor 文 404 中に Sub Directory 402 の中  
味が何処に記録されているかを示す File Entry 文 405  
の記録位置 (図20 (b) の例では6番目の論理プロ  
ック) が Long Allocation Descriptor 文で記載 (LAD  
2) している。

【0094】・論理ブロック番号“2”の論理ブロック  
に Sub Directory 402 の中味が記録されている位置を  
示す File Entry 文 405 が記録されている。

【0095】図20 (a) の例では Sub Directory 402  
の中には File Data 403 のみが入っている。Sub  
Directory 402 の中味が実質的には File Data 403  
に関する情報が記録されている File Identifier Des  
criptor 文 406 の記録位置を示す事になる。

【0096】File Entry 文中の Short Allocation Des  
criptor 文で3番目の論理ブロックに SubDirectory 40  
2 の中味が記録されている事 (AD(3)) が記録されて  
いる。

【0097】・論理ブロック番号“3”の論理ブロック  
に Sub Directory 402 の中味が記録されている。

【0098】図20 (a) の例では Sub Directory 402  
の中には File Data 403 のみが入っている。Sub  
Directory 402 の中味として File Data 403 に関する  
情報が File Identifier Descriptor 文 406 で記載  
されている。また図示して無いが同一論理ブロック内に  
Sub Directory402 自身の情報も File Identifier Des  
criptor 文で並記して有る。

【0099】File Data 403 に関する File Identifier  
Descriptor 文 406 の中にそのFile Data 403 の内  
容が何処に記録されている位置を示す File Entry 文 4  
07の記録位置 (図20 (b) の例では4番目の論理プロ  
ックに記録されている) がLong Allocation Descriptor  
文で記載 (LAD(4)) されている。

【0100】・論理ブロック番号“4”の論理ブロック  
に File Data 403 内容408、409が記録されている位置  
を示す File Entry 文 407 が記録されている。

【0101】File Entry 文 407 内の Short Allocatio  
n Descriptor 文で File Data 403内容408、409が5番  
目と6番目の論理ブロックに記録している事が記述 (A  
D(5),AD(6)) されている。

【0102】・論理ブロック番号“5”の論理ブロック  
に File Data 403 内容情報(a)408が記録されている。  
【0103】・論理ブロック番号“6”の論理ブロック  
に File Data 403 内容情報(b)409が記録されている。  
【0104】<図20 (b) 情報に沿った File Data  
へのアクセス方法>

<“情報記憶媒体上のファイルシステム情報記録内容  
>”で簡単に説明したように File Identifier Descrip  
tor 404、406 と File Entry 405、407 には、それに統

1 図の ルートディレクトリ (Root Directory) 401  
が存在し、その下にサブディレクトリ (SubDirector  
y) 402 が属している。この SubDirectory 402 の中に  
ファイルデータ (File Data) 403 が存在している。  
【0083】実際にはこの例に限らず Root Directory  
401 の直接下に File Data 403 が存在したり、複数の  
SubDirectory 402 が直列につながった複雑な階層構造  
を持つ場合もある。

【0084】<情報記憶媒体上ファイル管理情報の記録  
内容>ファイル管理情報は上述した論理ブロック単位で  
記録される。各論理ブロック内に記録される内容は以下  
の通りである。

【0085】・ファイルに関する情報を示す記述文 F  
ID (File Identifier Descriptor)  
ファイルの履歴やファイル名 (Root Directory 名、Su  
bDirectory 名、FileData 名など) を記述している。

【0086】FIDの中にそれに続く File Data のデ  
ータ内容や、Directory の中味の記録場所を示す記述文  
(つまり該当ファイルに対応した以下に説明する FE  
の記録位置も記述されている)。

【0087】・ファイル中味の記録位置を示す記述文 F  
E (File Entry)

File Data のデータ内容や、Directory (SubDirectory  
など) の中味に関する情報が記録されている情報記憶  
媒体上の位置 (論理ブロック番号) などを記述してい  
る。

【0088】File Identifier Descriptor の記述内容  
の抜粋を図25に示した。またその詳細の説明は<<Fi  
le Identifier Descriptor>>で行う。File Entry の  
記述内容の抜粋は図24に示し、その詳細な説明は<<  
File Entry>>で行う。

【0089】情報記憶媒体上の記録位置を示す記述文は  
図21に示す ロングアロケーションディスクリプター  
(Long Allocation Descriptor) と図22に示す ショ  
ートアロケーションディスクリプター (Short Allocati  
on Descriptor) を使っている。それぞれの詳細説明は  
<Long Allocation Descriptor>と<Short Allocation  
Descriptor>で行う。

【0090】例として図20 (a) のファイルシステム  
構造の情報を情報記憶媒体に記録した時の記録内容を図  
20 (b) に示す。図20 (b) の記録内容は以下の通  
りとなる。

【0091】・論理ブロック番号“1”の論理ブロック  
に Root Directory 401 の中味が示されている。

【0092】図20 (a) の例では Root Directory 40  
1 の中には Sub Directory402 のみが入っている。で  
、Root Directory 401 の中味として Sub Directory 402  
に関する情報が File Identifier Descriptor 文 404で  
記載されている。また図示して無いが同一論理ブロック内  
に Root Directory 401 自身の情報も File Identifier

(9) 特開2003-114817 16

(File Allocation Table) の記録場所はあらかじめ決まっているので記録媒体の高い信頼性 (欠陥領域が少ない事) が前提となる。

[0113] UDFではファイル管理情報が分散配置されているので、ファイル構造の大幅な変更が少なく、階層の下部分 (主に Root Directory より下の部分) で後から新たなファイル構造を付け足して行く用途 (主に追記用途) に適している。(追記時には以前のファイル管理情報に対する変更箇所が少ないため。) また分散されたファイル管理情報の記録位置を任意に指定できるの、先天的な欠陥箇所を避けて記録する事が出来る。

[0114] ファイル管理情報を任意の位置に記録できるもので全ファイル管理情報を一箇所に集めて記録し上記FATの利点も出せるので、より汎用性の高いファイルシステムと考えることが出来る。

[0115] <<UDFの各記述文 (Descriptor) の具体的な内容説明>>>

<<論理ブロック番号の記述文>>

<Allocation Descriptor>

“<情報記録媒体上のファイルシステム情報記録内容>” に示したように File Identifier Descriptor や File Entry などの一部に含まれ、その後に続く情報が記録されている位置 (論理ブロック番号) を示した記述文をAllocation Descriptorと呼ぶ。Allocation Descriptor には以下に示すLong Allocation Descriptor と Short Allocation Descriptor がある。

[0116] <Long Allocation Descriptor>

図21に示すように

- Extent の長さ 410 ... 論理ブロック数を 4Bytes で表示
- Extent の位置 411 ... 該当する論理ブロック番号を 4Bytes で表示
- Implementation Use 412 ... 演算処理に利用する情報で 8Bytes で表示などから構成される。

[0117] この説明文では記述を簡素化して “LAD (論理ブロック番号)” で記述する。

[0118] <Short Allocation Descriptor>

図22に示すように

- Extent の長さ 410 ... 論理ブロック数を 4Bytes で表示
- Extent の位置 411 ... 該当する論理ブロック番号を 4Bytes で表示

のみで構成される。

[0119] この説明文では記述を簡素化して “AD (論理ブロック番号)” で記述する。

[0120] <<Unallocated Space Entry>>図23に示すように情報記録媒体上の “未記録状態の Extent 分布” をExtent毎に Short Allocation Descriptor で記述し、それを並べる記述文で、SpaceTable (図27～図29参照) に用いられる。具体的な内容としては以

く情報が記述して有る論理ブロック番号が記述して有る。Root Directory から階層を下りながら SubDirectory を経由して File Data へ到達すると同様、File Identifier Descriptor と File Entry 内に記述して有る論理ブロック番号に従って情報記録媒体上の論理ブロック内の情報を順次再生しながらFile Data のデータ内容へアクセスする。

[0105] つまり図20 (b) に示した情報に対して File Data 403へアクセスするには、まず始めに1番目の論理ブロック情報を読む。File Data 403は Sub Directory 情報の中から Sub Directory 402の File Identifier Descriptor 404を探し、LAD(2)を読み取った後、それに従って2番目の論理ブロック情報を読む。2番目の論理ブロックには1個の File Entry 文が記述してないので、その中の AD(3)を読み取り、3番目の論理ブロックへ移動する。3番目の論理ブロックでは File Data 403に関して記述して有る File Identifier Descriptor 406を探し、LAD(4)を読み取る。LAD(4)に従い4番目の論理ブロックへ移動すると、そこには1個の File Entry 文 407しか記述してないので、AD(5)とAD(6)を読み取り、File Data 403の内容が記録して有る論理ブロック番号 (5番目と6番目) を見付け、[0106] AD (\*), LAD (\*) の内容については “<UDFの各記述文 (Descriptor) の具体的な内容説明>” で詳細に説明する。

[0107] <<UDFの特徴>>

<UDFの特徴説明>以下にHDDやFDD、MOなどで使われているFATとの比較によりUDFの特徴を説明する。

[0108] 1) (最小論理ブロックサイズ、最小論理セクタサイズなどの) 最小単位が大きく、記録すべき情報量の多い映像情報や音楽情報の記録に向く。

[0109] FATの論理セクタサイズが512 Bytes に対して、UDFの論理セクタ (ブロック) サイズは2048 Bytes と大きくなっている。

[0110] 2) FATはファイルの情報記録媒体への割り当て管理表 (File Allocation Table) が情報記録媒体上で局所的に集中記録されるのに対し、UDFではファイル管理情報をディスク上の任意の位置に分散記録できる。

[0111] UDFではファイル管理情報やファイルデータに関するディスク上の記録位置は論理セクタ (ブロック) 番号として Allocation Descriptor に記述される。

[0112] FATではファイル管理領域 (File Allocation Table) で集中管理されているため頻繁にファイル構造の変更が必要な用途 (主に頻繁な書き換え用途) に適している。(集中箇所に記録されているので管理情報を書き換え易いため。) またファイル管理情報

(10) 特開2003-114817 18

[0133] 図26のファイルシステム構造の情報を含むDFフォーマットに従って情報記録媒体上に記録した例を図27～図29に示す。

[0134] 情報記録媒体上の未記録位置管理方法として次のようなものがある。

[0135] • Space Bitmap 方法

Space Bitmap Descriptor 470を用いた、情報記録媒体内記録領域の全論理ブロックに対してビットマップ的に “記録済み” または “未記録” のフラグを立てる。

[0136] • Space Table 方法

Unallocated Space Entry 471の記述方式を用いて Short Allocation Descriptor の列記として未記録の全論理ブロック番号を記録している。

[0137] 本実施形態では説明上、わざと図27～図29に両方式を併記しているが、実際には両方が一緒に使われる (情報記録媒体上に記録される) ことはほとんど無く、どちらか一方のみ使われている。

[0138] 図27～図29に記述されている主な Descriptor の内容の概説は以下の通りである。

[0139] • Bagging Extended Area Descriptor 445 ... Volume Recognition Sequenceの開始位置を示す

- Volume Structure Descriptor 446 ... Volume の内容説明を記述。
- [0140] • Boot Descriptor 447 ... ブート時の処理内容を記述。
- [0141] • Terminating Extended Area Descriptor 448 ... Volume Recognition Sequence の終了位置を示す。

[0142] • Partition Descriptor 450 ... パーティション情報 (サイズなど) を示すDVD-RAMでは1Volume 当たり1Partition を原則としている。

[0143] • Logical Volume Descriptor 454 ... 論理ボリュームの内容を記述している。

[0144] • Anchor Volume Descriptor Pointer 458 ... 情報記録媒体記録領域内でのMain Volume Descriptor Sequence 449とMain Volume Descriptor Sequence 460の記録位置を示している。

[0145] • Reserved (all 00h bytes) 459 ~ 465 ... 特定の Descriptor を記録する論理セクタ番号を確保するため、その間に全て “0” を記録した調整領域を持たせている。

[0146] • Reserve Volume Descriptor Sequence 467 ... Main Volume Descriptor Sequence 449に記録された情報のバックアップ領域。

[0147] <<再生時のファイルデータへのアクセス方法>>図27～図29に示したファイルシステム情報を用いて例えば File Data H432 のデータ内容が再生するための情報記録媒体上のアクセス処理方法について説明する。

下に表示ものが記述されている。

[0121] • Descriptor Tag 413 ... 記述内容の識別子を表し、この場合は “263”

• ICB Tag 414 ... ファイルタイプを示す

ICB Tag 内の File Type=1 は Unallocated Space Entry を意味し、File Type=4 は Directory、File Type=5 は File Data を表している。

[0122] • Allocation Descriptors 列の全長 415 ... 4Bytes で総 Bytes 数を示す。

[0123] <<File Entry>>

“情報記録媒体上のファイルシステム情報記録内容” で説明した記述文であり、図24に示すように以下に示すものが記述されている。

[0124] • Descriptor Tag 417 ... 記述内容の識別子を表し、この場合は “261”

• ICB Tag 418 ... ファイルタイプを示す → 内容は前述の ICB Tag 414と同じ

• Permissions 419 ... ユーザ別の記録・再生・削除許可情報を示す。主にファイルのセキュリティ確保を目的として使われる。

[0125] • Allocation Descriptors 420 ... 該当ファイルの意味が記録して有る位置をExtent 毎にShort Allocation Descriptor を並べて記述する。

[0126] <<File Identifier Descriptor>>

“情報記録媒体上のファイルシステム情報記録内容” で説明したようにファイル情報を記述した記述文であり、図25に示すように以下に示すものが記述されている。

[0127] • Descriptor Tag 421 ... 記述内容の識別子を表し、この場合は “257”

• File Characteristics 422 ... ファイルの種類を示し、Parent Directory、Directory、File Data、File Entry、Parent Directory、Directory、File Data、File Entryの削除フラグのどれかを意味する。

[0128] • Information Control Block 423 ... このファイルに対応したF/E位置がLong Allocation Descriptor で記述されている。

[0129] • File Identifier 424 ... ディレクトリ名またはファイル名。

[0130] • Padding 437 ... File Identifier Descriptor 全体の長さを調整するために付加されたダミー領域で、通常は全て “0” が記録されている。

[0131] <<UDFに比べて情報記録媒体上に記録したファイル構造記述例>>>

“<<UDFの概要>>” で示した内容について具体的な例を用いて以下に詳細に説明する。

[0132] 図20 (a) に対して、より一般的なファイルシステム構造例を図26に示す。括弧内は Directory の中身に関する情報または File Data のデータ内容が記録されている情報記録媒体上の論理ブロック番号を示している。

【0148】1) 情報記録再生装置起動時または情報記憶媒体装着時のブート (Boot) 領域として Volume Recognition Sequence 444 領域内の Boot Descriptor 447 の情報を再生し、Boot Descriptor 447 の記述内容に沿ってブート (Boot) 時の処理が始まる。

【0149】2) 特に指定されたブート時の処理が無い場合には、始めに Main Volume Descriptor Sequence 449 領域内の Logical Volume Descriptor 454 の情報を再生する。

【0150】3) Logical Volume Descriptor 454 の中に Logical Volume Contents Use 455が記述されており、そこに File Set Descriptor 472 が記録して有る位置を示す論理ブロック番号が Long Allocation Descriptor (図21) 形式で記述して有る。(図27～図29の例ではLAD(100)から100番目の論理ブロックに記録して有る。)

【0151】4) 100番目の論理ブロック (論理セクタ番号では372番目になる) にアクセスし、File Set Descriptor 472を再生する。その中のRoot Directory 108 473にRoot Directory A 425に関するFile Entry が記録されている場所 (論理ブロック番号) が Long Allocation Descriptor (図21) 形式で記述して有る。(図27～図29の例ではLAD(102)から102番目の論理ブロックに記録して有る。)

【0151】5) Root Directory 108 473 のLAD(102)に従い、102番目の論理ブロックにアクセスし、Root Directory A 425に関するFile Entry 475を再生し、Root Directory A 425の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(103))。

【0152】6) 103番目の論理ブロックにアクセスし、Root Directory A 425の中身に関する情報を再生する。File Data H 432はDirectory D 428系列の下に存在するので、Directory D 428に関するFile Identifier Descriptor を探し、SubDirectory F 430に関するFile Entry が記録して有る位置 (論理ブロック番号) を読み込む (AD(103))。

【0153】7) 110番目の論理ブロックにアクセスし、Directory D 428の中身に関するFile Entry 480を再生し、Directory D 428の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(111))。

【0154】8) 111番目の論理ブロックにアクセスし、Directory D 428の中身に関する情報を再生する。File Data H 432はSubDirectory F 430の直接下に存在するので、SubDirectory F 430に関するFile Identifier Descriptor を探し、SubDirectory F 430に関するFile Entry が記録して有る位置 (論理ブロック番号) を読み込む (AD(111))。

【0159】11) 111番目の論理ブロックにア

AD(112)) を読み取る。

【0155】9) 112番目の論理ブロックにアクセスし、SubDirectory F 430に関するFile Entry 482を再生し、SubDirectory F 430の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(113))。

【0156】10) 113番目の論理ブロックにアクセスし、SubDirectory F 430の中身に関する情報を再生し、File Data H 432に関するFile Identifier Descriptor を探し、File Data H 432の中身に関するFile Entry が記録して有る位置 (論理ブロック番号) を読み込む (AD(114))。

【0157】11) 114番目の論理ブロックにアクセスし、File Data H 432に関するFile Entry 484を再生し、File Data H 432のデータ内容 489が記録されている位置を読み取る。

【0158】12) File Data H 432に関するFile Entry 484内に記述されている論理ブロック番号順に情報記憶媒体から情報を再生して File Data H 432のデータ内容 489を読み取る。

【0159】<<<特定のファイルデータ内容変更方法>>>図27～図29に示したファイルシステム情報を用いて例えば File Data H432のデータ内容を変更する場合のアクセスも含めた処理方法について説明する。

【0160】1) File Data H 432の変更前後のデータ内容の容量差を求め、その値を2048 Bytesで割り、変更後のデータを記録するのに論理ブロック何個追加使用するかまたは何個不要になるかを事前に計算しておく。

【0161】2) 情報記録再生装置起動時または情報記憶媒体装着時のブート (Boot) 領域として Volume Recognition Sequence 444 領域内の Boot Descriptor 447 の情報を再生し、Boot Descriptor 447 の記述内容に沿ってブート (Boot) 時の処理が始まる。

【0162】3) 特に指定されたブート時の処理が無い場合には、始めに Main Volume Descriptor Sequence 449 領域内の Partition Descriptor 450を再生し、その中に記述して有る Partition Contents Use 451の情報を読み取る。この PartitionContents Use 451 (Partition Header Descriptor と呼ぶ) の中に Space Table もしくは Space Bitmap の記録位置が示して有る。Space Table 位置は Unallocated Space Table 452 の欄に Short Allocation Descriptor の形式で記述されている。(図27～図29の例ではAD(50)) また、Space Bitmap 位置は Unallocated Space Bitmap 453の欄に Short Allocation Descriptor の形式で記述されている。(図27～図29の例ではAD(0))

4) 3) で読み取った Space Bitmap が記述して有る論理ブロック番号 (0) へアクセスする。Space Bitmap D

descriptor 470 から Space Bitmap 情報を読み取り、未記録の論理ブロックを探し、1) の計算結果分の論理ブロックの使用を登録する (Space Bitmap Descriptor 460 情報の書き換え処理)。

【0163】4') もしくは、3) で読み取った Space Table が記述して有る論理ブロック番号 (50) へアクセスする。Space Table の USE(AD(\*), AD(\*), ..., AD(\*)) 471から未記録の論理ブロックを探し、1) の計算結果分の論理ブロックの使用を登録する (Space Table 情報の書き換え処理)。

【0164】5) 次に Main Volume Descriptor Sequence 449 領域内の Logical Volume Descriptor 454の情報を再生する。

【0165】6) Logical Volume Descriptor 454の中に Logical Volume Contents Use 455が記述されており、そこに File Set Descriptor 472 が記録して有る位置を示す論理ブロック番号が Long Allocation Descriptor (図21) 形式で記述して有る。(図27～図29の例ではLAD(100)から100番目の論理ブロックに記録して有る。)

【0166】7) 100番目の論理ブロック (論理セクタ番号では400番目になる) にアクセスし、File Set Descriptor 472を再生する。その中のRoot Directory 108 473にRoot Directory A 425に関するFile Entry が記録されている場所 (論理ブロック番号) が Long Allocation Descriptor (図21) 形式で記述して有る。(図27～図29の例ではLAD(102)から102番目の論理ブロックに記録して有る。)

【0167】8) Root Directory 108 473 のLAD(102)に従い、102番目の論理ブロックにアクセスし、Root Directory A 425に関するFile Entry 475を再生し、Root Directory A 425の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(103))。

【0167】9) 103番目の論理ブロックにアクセスし、Root Directory A 425の中身に関する情報を再生する。File Data H 432はDirectory D 428系列の下に存在するので、Directory D 428に関するFile Identifier Descriptor を探し、SubDirectory F 430に関するFile Entry が記録して有る位置 (論理ブロック番号) を読み込む (AD(110))。

【0168】10) 110番目の論理ブロックにアクセスし、Directory D 428の中身に関するFile Entry 480を再生し、Directory D 428の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(111))。

【0169】11) 111番目の論理ブロックにア

スし、Directory D 428の中身に関する情報を再生する。File Data H 432はSubDirectory F 430の直接下に存在するので、SubDirectory F 430に関するFile Identifier Descriptor を探し、SubDirectory F 430に関するFile Entry が記録して有る位置 (論理ブロック番号) を読み取る。

【0170】12) 112番目の論理ブロックにアクセスし、SubDirectory F 430に関するFile Entry 482を再生し、SubDirectory F 430の中身に関するFile Entry が記録して有る位置 (論理ブロック番号) を読み込む (AD(113))。

【0171】13) 113番目の論理ブロックにアクセスし、SubDirectory F 430の中身に関する情報を再生し、File Data H 432に関するFile Identifier Descriptor を探し、File Data H 432の中身に関するFile Entry が記録して有る位置 (論理ブロック番号) を読み取る。

【0172】14) 114番目の論理ブロックにアクセスし、File Data H 432に関するFile Entry 484を再生し、File Data H 432のデータ内容 489が記録されている位置を読み取る。

【0173】15) 4) か4') で追加登録した論理ブロック番号も加味して変更後のFile Data H 432のデータ内容 489を記録する。

【0174】<<<特定のファイルデータ/ディレクトリ消去処理方法>>>例としてFile Data H 432またはSubDirectory F 430を消去する方法について説明する。

【0175】1) 情報記録再生装置起動時または情報記憶媒体装着時のブート (Boot) 領域として Volume Recognition Sequence 444 領域内の Boot Descriptor 447 の情報を再生し、Boot Descriptor 447 の記述内容に沿ってブート (Boot) 時の処理が始まる。

【0176】2) 特に指定されたブート時の処理が無い場合には、始めに Main Volume Descriptor Sequence 449 領域内の Logical Volume Descriptor 454の情報を再生する。

【0177】3) Logical Volume Descriptor 454の中に Logical Volume Contents Use 455が記述されており、そこに File Set Descriptor 472 が記録して有る位置を示す論理ブロック番号が Long Allocation Descriptor (図21) 形式で記述して有る。(図27～図29の例ではLAD(100)から100番目の論理ブロックに記録して有る。)

【0178】4) 100番目の論理ブロック (論理セクタ番号では400番目になる) にアクセスし、File Set Descriptor 472を再生する。その中のRoot Directory 108 473にRoot Directory A 425に関するFile Entry が記録さ

れている場所 (論理ブロック番号) が Long Allocation Descriptor (図21) 形式で記述して有る (図27～図29の例ではLAD(102)から102番目の論理ブロックに記録して有る)。

【0178】5) Root Directory 108 473 のLAD(102)に従い、102番目の論理ブロックにアクセスし、Root Directory A 425 に関するFile Entry 475 を再生し、Root Directory A 425 の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(103))。

【0179】6) 103番目の論理ブロックにアクセスし、Root Directory A 425 の中身に関する情報を再生する。File Data H 432 は Directory D 428 系列の下に存在するので、Directory D 428 に関するFile Identifier Descriptor を探し、Directory D 428 に関するFile Entry が記録して有る論理ブロック番号 (図27～図29には図示して無いがLAD(100)) を読み取る。

【0180】7) 110番目の論理ブロックにアクセスし、Directory D 428 に関するFile Entry 480 を再生し、Directory D 428 の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(111))。

【0181】8) 111番目の論理ブロックにアクセスし、Directory D 428 の中身に関する情報を再生する。File Data H 432 は SubDirectory F 430 の直接下に存在するので、SubDirectory F 430 に関するFile Identifier Descriptor を探す。

【0182】SubDirectory F 430 を消去する場合には、SubDirectory F 430 に関するFile Identifier Descriptor 内のFile Characteristics 422 (図25) に“ファイル削除フラグ”を立てる。SubDirectory F 430 に関するFile Entry が記録して有る論理ブロック番号 (図27～図29には図示して無いがLAD(112)) を読み取る。

【0183】9) 112番目の論理ブロックにアクセスし、SubDirectory F 430 に関するFile Entry 482 を再生し、SubDirectory F 430 の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(113))。

【0184】10) 113番目の論理ブロックにアクセスし、SubDirectory F 430 の中身に関する情報を再生し、File Data H 432 に関するFile Identifier Descriptor を探す。

【0185】File Data H 432 を消去する場合には、File Data H 432 に関するFile Identifier Descriptor 内のFile Characteristics 422 (図25) に“ファイル削除フラグ”を立てる。さらにそこからFile Data H 432 に関するFile Entry が記録して有る論理ブロック番号 (図27～図29には図示して無いがLAD

(114)) を読み取る。  
【0186】11) 114番目の論理ブロックにアクセスし、File Data H 432 に関するFile Entry 484 を再生し、File Data H 432 のデータ内容 489 が記録されている位置を読み取る。

【0187】File Data H 432 を消去する場合には、以下の方法でFile Data H 432 のデータ内容 489 が記録されている論理ブロックを解放する (その論理ブロックを未記録状態に型録する)。

【0188】12) 次に Main Volume Descriptor Sequence 449 領域内の Partition Descriptor 450 を再生し、その中に記述して有る Partition Contents Use 451 の情報を読み取る。この Partition Contents Use 451 (Partition Header Descriptor と呼ぶ) の中にSpace Table もしくは Space Bitmap の記録位置が示して有る。

【0189】Space Table 位置は Unallocated Space Identifier Descriptor の形式で記述されている。 (図27～図29の例ではAD(50)) また、Space Bitmap 位置は Unallocated Space Identifier Descriptor の中に記述されている。 (図27～図29の例ではAD(50))

【0190】13) もしくは、12) で読み取ったSpace Table を Space Table に書き換える。

【0191】13') もしくは、12) で読み取ったSpace Table を Space Table に書き換える。  
【0191】実際の処理は“13)”か“13')”のうちのどちらか一方の処理を行う。

【0192】File Data H 432 を消去する場合には、12) 10)～11) と同じ手順を繰り返してFile Data H 433 のデータ内容 490 が記録されている位置を読み取る。

【0193】13) 次に Main Volume Descriptor Sequence 449 領域内の Partition Descriptor 450 を再生し、その中に記述して有る Partition Contents Use 451 の情報を読み取る。この Partition Contents Use 451 (Partition Header Descriptor と呼ぶ) の中にSpace Table もしくは Space Bitmap の記録位置が示して有る。

【0194】Space Table 位置は Unallocated Space Identifier Descriptor の形式で記述されている。 (図27～図29の例ではAD(50)) また、Space Bitmap 位置は Unallocated Space Identifier Descriptor の中に記述されている。 (図27～図29の例ではAD

のうちのどちらか一方の処理を行う。  
【0203】5) 次に Main Volume Descriptor Sequence 449 領域内の Logical Volume Descriptor 454 の情報を再生する。

【0204】6) Logical Volume Descriptor 454 の中にLogical Volume Contents Use 455が記述されておる。そこにFile Set Descriptor 472 が記録して有る位置を示す論理ブロック番号が Long Allocation Descriptor (図21) 形式で記述して有る。 (図27～図29の例ではLAD(100)から100番目の論理ブロックに記録して有る)。

【0205】7) 100番目の論理ブロック (論理セクタ番号では400番目になる) にアクセスし、File Set Descriptor 472 を再生する。その中のRoot Directory 108 473 にRoot Directory A 425 に関するFile Entry が記録されている場所 (論理ブロック番号) が Long Allocation Descriptor (図21) 形式で記述して有る (図27～図29の例ではLAD(102)から102番目の論理ブロックに記録して有る)。

【0205】8) Root Directory 108 473 のLAD(102)に従い、102番目の論理ブロックにアクセスし、Root Directory A 425 に関するFile Entry 475 を再生し、Root Directory A 425 の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(103))。

【0206】9) 103番目の論理ブロックにアクセスし、Root Directory A 425 の中身に関する情報を再生する。

【0207】Directory D 428 に関するFile Identifier Descriptor を探し、Directory D 428 に関するFile Entry が記録して有る論理ブロック番号 (図27～図29には図示して無いがLAD(110)) を読み取る。

【0208】10) 110番目の論理ブロックにアクセスし、Directory D 428 に関するFile Entry 480 を再生し、Directory D 428 の中身に関する情報が記録されている位置 (論理ブロック番号) を読み込む (AD(111))。

【0209】11) 111番目の論理ブロックにアクセスし、Directory D 428 の中身に関する情報を再生する。

【0210】SubDirectory F 430 に関するFile Identifier Descriptor を探し、SubDirectory F 430 に関するFile Entry が記録して有る論理ブロック番号 (図27～図29には図示して無いがLAD(112)) を読み取る。

【0211】12) 112番目の論理ブロックにアクセスし、SubDirectory F 430 に関するFile Entry 482 を再生し、SubDirectory F 430 の中身に関する情報が記録されている位置 (論理ブロック番号) を読み





レス空間上で各種ファイルの配置設定方法と比較した図である。

【図3】図1に示す共有方法において、UDF論理アドレス空間上のDVDオブジェクトファイルの配置設定方法の一例を示す図である。

【図4】図1に示す共有方法において、UDF論理アドレス空間上のCE領域に格納したファイルシステム空間を示すファイルの配置設定方法の一例を示す図である。

【図5】図1に示す共有方法における各種ファイルと擬似ファイルの説明するための図である。

【図6】CE\_FILE\_AREA、CEFFのファイルサイズの拡大(縮小)を説明するための図である。

【図7】UDFによる従来のボリューム&ファイルストラクチャの読み出し手順を示すフローチャートである。

【図8】図1に示す共有方法を採用したときのボリューム&ファイルストラクチャの読み出し手順を示すフローチャートである。

【図9】CE\_FILE\_AREA、CEFFの割り当てと、WRITE/READ処理方法を説明するための図である。

【図10】Triggerファイルのデータ構造の一例を示す図である。

【図11】TriggerファイルによりCEファイル位置を確認する処理を示すフローチャートである。

【図12】Triggerファイルの先頭セクタが読み出せない場合の処理を示すフローチャートである。

【図13】Triggerファイルを活用しCEファイルを追加又は削除する処理を示すフローチャートである。

【図14】図1に示す共有方法を実現するためのシステムの概略構成を示す図である。

【図15】図1に示す共有方法とは異なる共有方法を示す図である。

【図16】図15に示す共有方法におけるCE領域に格納したファイルシステム空間上のUDF論理アドレス空間を示すファイル配置を示す図である。

【図17】Triggerファイルで管理されるCE\_FILE\_AREA、CEFFへの記録処理を示すフローチャートである。

【図18】Triggerファイルで管理されるCE\_FILE\_AREA、CEFFの追加処理を示すフローチャートである。

ャートである。

【図19】Triggerファイルで管理されるCE\_FILE\_AREA、CEFFのサイズ縮小処理を示すフローチャートである。

【図20】階層化されたファイルシステム構造と情報記憶媒体上へ記録された情報内容との間の基本的な関係の概念を示した概念説明図である。

【図21】Long Allocation Descriptor (Extent)の位置を示す大きいサイズの記述文)の記述内容説明図である。

【図22】Short Allocation Descriptor (Extent)の位置を示す小さいサイズの記述文)の記述内容説明図である。

【図23】Unallocated Space Entry (未記録なExtentの情報記憶媒体上の位置に関する特設登録用の記述文)の記述内容説明図である。

【図24】File Entry (Fileの属性とFileの記録位置の情報登録に関する記述文)の記述内容の一部抜粋した内容説明図である。

【図25】File Identifier Descriptor (Fileの名前と対応したFIEの記録位置に関する記述文)の記述内容の一部抜粋した内容説明図である。

【図26】ファイルシステム構造の一例を示す図である。

【図27】UDFに従って情報記憶媒体上にファイルシステムを記録した例を示す図の一部(その1)である。

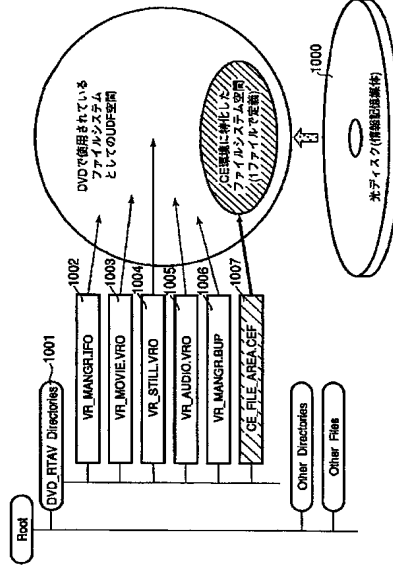
【図28】UDFに従って情報記憶媒体上にファイルシステムを記録した例を示す図の一部(その2)である。

【図29】UDFに従って情報記憶媒体上にファイルシステムを記録した例を示す図の一部(その3)である。

【符号の説明】

1001...DVD\_RTAV Directories  
1002...VR\_MANGR. IFO  
1003...VR\_MOVIE. VRO  
1004...VR\_STILL. VRO  
1005...VR\_AUDIO. VRO  
1006...VR\_MANGR. BUP  
1007...CE\_FILE\_AREA. CEF  
2001...TOC\_FILE. IFO  
2002...AV\_MANG. MNG  
2004...AV\_FILE\_01.MPG  
2005...AV\_FILE\_02.MPG  
2006...UDF\_FILE\_AREA. UDF

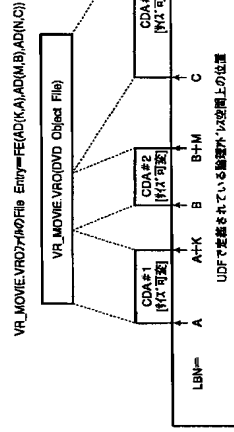
【図1】



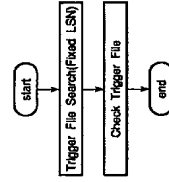
【図2】

比較項目	PG Fileと管理File	DVD Object File	CE領域に格納したファイルがUDF空間を表すファイル
CDA#1 (記録時の書き出し単位)	CDA設定無し 2KB単位で書き当て	2MB以上の可変	4MB固定
CDA境界位置の保持	CDA設定無し	境界位置を保持しない	常にCDA境界位置を保持する
部分洋装束の格納単位	2KB単位	2KB単位	4MB固定 (CDA単位の格納)
CDAとExtentの関係	CDA設定無し	1個以上のExtentで1個のCDAを格納	rCDA=1(Extent=1)を保持
図1での該当File	VR_MANGR. IFO VR_MANGR. BUP VR_AUDIO. VRO	VR_MOVIE. VRO VR_STILL. VRO VR_AUDIO. VRO	CE_FILE_AREA.CEF

【図3】



【図8】

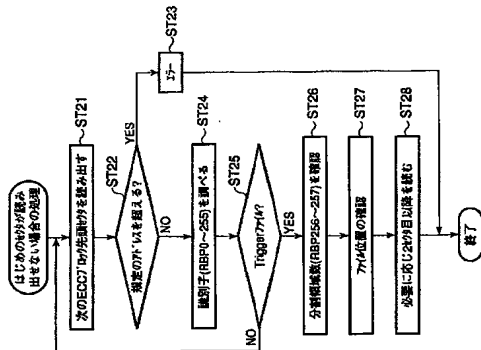




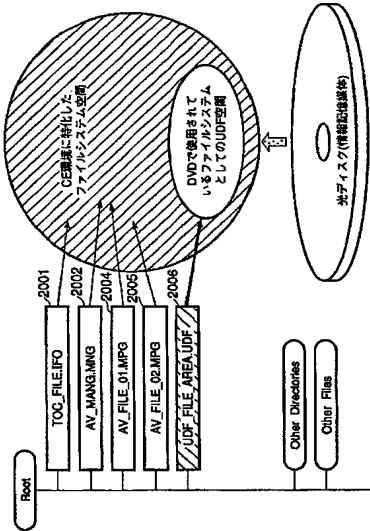
【图10】

RBP	Length	Contents
0 ~ 255	256	Trigger-Firmware識別子
256 to 257	2	予約領域
258 to 269	12	固定長CMAデータ(セクタ数)
260 to 261	2	変長固定長除算結果(固定長CMA単位0x100h) 例: CMA数は16個とするなら, 0x010h
262 to 265	4	1番目の領域開始LSN
266 to 269	4	1番目の領域長(セクタ数)
270 to 273	4	2番目の領域開始LSN
274 to 277	4	2番目の領域長(セクタ数)
...	...	...
N to N+3	4	n番目の領域開始LSN
N+4 to N+7	4	n番目の領域長(セクタ数)
N+8 to LN	Reserved	LN+8 ~ ECCプロロクの連結のRBP
LN+1 to ...	...	上記RBP ~ LNの内容を複数回繰り返す 領域に属するもの

【図12】



【图 15】



【23】

```
USE(AD(*),AD(*)...AD(*))
```

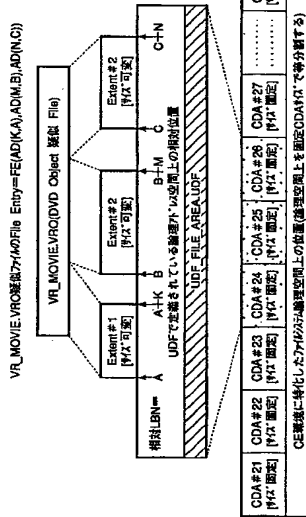
Descriptor	ICB Tag	Allocation Descriptors	Allocation Descriptors
ファイル名 (type=253)	ファイルのタイプを示す	列の全長 (Bytes)	位置的情報記述媒体上の処理位置
記述内容の識別子 413	1 (type=1)	415	(情報記述媒体上の処理) を並べて記記する
識別子 1 (type=1)	414	415	を並べて記記する
1 (type=1)	414	415	(AD*)...AD(*)

【☒24】

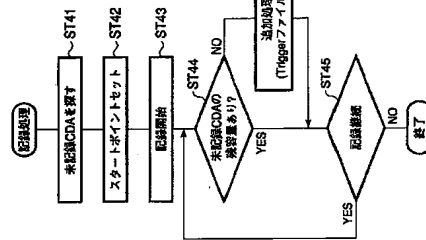
$$FE(AD(*), AD(*) \dots AD(*))$$

Descriptor	ICB Tag	Permissions	Allocation Descriptors
TagNo(=281)	ファイル名	ユーザ別の 記録・共有・ 削除	Fileの格納に帰属本体上記録 位置
TagNoInFile(=417)	ファイル内の データ番号	タイプ(=45) 418	(情報記憶媒体上の物理) (ロッキング番号) を並べて列記
FileBytes(=168bytes)	168bytes	許可情報 418 [32B/byte]	(AD)*(*)AD*(*)AD*(*)

【图16】



【图 17】



【25】

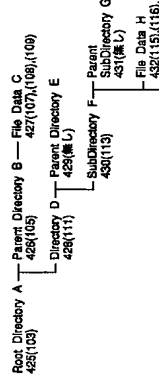
FID/LAD(論理ブロック番号))

Descriptor	File	Information	File	Padding
Length (n=257)	ファイルの識別	対応したBlock	ディレクトリ	パディング領域
総括内容子 421	をがす 422	配置位置 423	主ファイル名	(000h) 437
[16Byte]	[1Byte]	[LAD(*)]	データ名	437
			424	

【☒27】

LSN/LBN	Structure 441	Discipline 442	Contents 443
D-15		Reserved 459 (all 00n bytes)	
16	Volume Revision Sequence 444	Beginning Ext. Area Descr. 445 Volume Structure Descr. 446 Boot Descriptor 447 Terminating Ext. Area D. 448	VMS標誌位置 DISC管理說明 Boot標誌位置 VMS終了位置
17		Reserved 460 (all 00n bytes)	
18			
19			
~31		留稱	
32~			
34	Main Volume Descriptor Table 452 Anchor Point 449	Partition Descriptor 450 Partition Contents Use 451 Unallocated Space Table 453 Anchor Point 450 Unallocated Space Bitmap 455 AD(0)	Space Table の配置位置 Special Bitmap の配置位置
35		Logical Volume Descriptor 454 Logical Volume Cont. Use 455 LAD(103)	File Set Descriptorの 配置位置
~47		留稱	
~63			
~255		Reserved 461 (all 00n bytes)	
256	First Anchor Point 456	Anchor Volume Descriptor Pointer 458	
~271		Reserved 462 (all 00n bytes)	

【图26】



【図28】

372 0	Space Bitmap Descriptor 470	Space Bitmap 470
372 49	USE(AD(*),AD(*))*,...	Space Table 471
372 80	AD(*)471	Extents 472
371 89	File Set Descriptor 472	Root Directory 473
372 100	Root Directory 473	FEの記録位置 474
373 101	LAD(102) 474	475
374 102	Root Directory AFE AD(103)475	FIDの記録位置 476
375 103	A FID(AD(104), LAD(110)476	B,DFE位置 477
376 104	Parent Direct. BFE(AD(105)477	FIDの記録位置 478
377 105	BOF(AD(106)478	COFFE位置 479
378 106	FE(AD(107)AD(108) AD(109)479	FileData位置 480
382 110	DirectoryD FE(AD(111)480	FIDの記録位置 481
383 111	D FID(AD(112), LAD(113)481	EFFE位置 482
384 112	SubDirectoryF FE(AD(113)482	FIDの記録位置 483
385 113	FID(AD(114) LAD(115)483	H,IFE位置 484
386 114	FE(AD(115)AD(116) AD(117)484	FileData位置 485
390 118	TFE(AD(118)AD(120) 485	FileData位置 486

【図29】

379 107	File Data 487	File Data CO情報 488
387 116	File Data 487	File Data HO情報 489
391 119	File Data 487	File Data IO情報 490
LSN-271 ~LSN-287	Reserved 483 (all 00h bytes)	
LSN-255	Second Anchor Point 487	Anchor Volume Descriptor Pointer 488
LSN-255 ~LSN-224	Reserved 484 (all 00h bytes)	
LSN-223 ~LSN-208	Reserved Volume Descriptor Sequence 487	Partition Contents Use Unallocated Space Table Sequence 488
LSN-207 ~LSN	Reserved 485 (all 00h bytes)	Main Volume Descriptor Sequence backup

フロントページの続き

(72) 発明者 三村 英紀  
神奈川県川崎市幸区柳町70番地 株式会社  
東芝柳町事業所内

Fターム(参考) 5B082 EA01 JA01  
5C052 AA04 AB09 CC06  
5D044 AB05 AB07 BC02 CC04 DE03  
5D090 AA01 CC14 GG21